



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/713,623	11/13/2003	Christian Behrens	16104-009001 / 2003P00802	2446
32864	7590	10/16/2007	EXAMINER	
FISH & RICHARDSON, P.C. PO BOX 1022 MINNEAPOLIS, MN 55440-1022			TIMBLIN, ROBERT M	
			ART UNIT	PAPER NUMBER
			2167	
			MAIL DATE	DELIVERY MODE
			10/16/2007	PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

# Office Action Summary

Application No.

10/713,623

Applicant(s)

BEHRENS ET AL.

Examiner

Robert M. Timblin

Art Unit

2167

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 01 August 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-7,9-12 and 15-25 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-7,9-12 and 15-25 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_

### **DETAILED ACTION**

This Office Action corresponds to application 10/713,623 and applicant's remarks filed August 1, 2007.

#### ***Response to Amendment***

New claims 24-25 have been added. Accordingly claims 1-7, 9-12, and 15-25 are pending.

#### ***Claim Rejections - 35 USC § 112***

The prior rejections under 35 USC 112 have been withdrawn in light of the amendments.

#### ***Claim Rejections - 35 USC § 102***

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-7, 9-12, and 15-13 are rejected under 35 U.S.C. 102(b) as being anticipated by Zhou et al ('Zhou' hereinafter) U.S. Patent Application 2002/0162093 A1.

With respect to claim 1, Zhou teaches a computing system comprising:

a user interface (102 (1-n)) configured to collect multiple data entries from a corresponding user device (figure 1, abstract, 0021; requests);

an object model controller (drawing references 220, 222, and figure 2) configured to associate, at runtime (abstract), the multiple data entries (figure 1; requests) with an object belonging to a model class (paragraphs 0036-0038), and to prevent the object from being modified by another user interface (0102);

business logic (drawing reference 204, and paragraphs 0030-0034) configured to process objects belonging to the model class (paragraphs 0036-0038); and

an intermediate layer (drawing reference 202 and figure 2) interposed between the user interface (at least drawing references 110, 212 and figures 1-2) and object model controller (drawing references 220, 222, and figure 2), and the business logic (drawing reference 204);

wherein the object model controller (drawing references 220, 222, and figure 2) is configured to provide the object with which the multiple data entries are associated to the intermediate layer, and wherein the intermediate layer is configured to rearrange data in the object into a format that is optimized for processing by the business logic and provide the object whose data has been rearranged to the business logic for processing (paragraphs 0031-0033).

With respect to claim 2, Zhou teaches the computing system of claim 1 wherein the system is configured to conduct a data flow between the user interface and the business logic through the intermediate layer (figure 2 and 3 and paragraphs 0038 and 0057).

With respect to claim 3, Zhou teaches the computing system of claim 2 wherein the data flow is initiated by one or more actions of the user interface, wherein the one or more actions comprise any one of an opening of a user interface and an entering of data in the user interface (figure 4, drawing references 400, 406, 408).

With respect to claim 4, Zhou teaches the computing system of claim 1 wherein the intermediate layer is further configured to optimize the arrangement of data for the business logic

(0031), wherein the rearrangement of data collected by the user interface comprises data collection (paragraph 0076) from the user interface and translating the collected data for the business logic (paragraph 0033).

With respect to claim 5, Zhou teaches the computing system of claim 1 wherein the intermediate layer is configured to provide a buffering of data flow between the user interface and the business logic, wherein the buffering of data flow enables the system to perform batch processing of a plurality of business processes (0074 and 0085; resource bundle).

With respect to claim 6, Zhou teaches the computing system of claim 1 wherein the business logic comprises a general business logic layer for common business functions and applications, wherein the intermediate layer is further configured to format the data for use in the general business logic layer (figure 4 and business logic layer 204).

With respect to claim 7, Zhou teaches the computing system of claim 1 wherein the intermediate layer is configured to perform one or more operations on one or more objects to reduce an amount of business processes performed by the business logic (paragraph 0030), wherein the one or more operations on the one or more objects comprise collecting and formatting one or more classes of objects (abstract, 0022 and 0047).

With respect to claim 9, Zhou teaches The computing system of claim 1 wherein the object model controller is configured to send data requests to the intermediate layer (figure 1),

Art Unit: 2167

wherein the data requests comprise any one of a read data request, a modify data request, and an insert data request, and wherein the object model controller further comprises an object-oriented interface (paragraph 0051).

With respect to claim 10, Zhou teaches the computing system of claim 1 further comprising a database configured to receive data from the business logic and send data to the business logic (paragraph 0026 and drawing references 108 424, and 432) and wherein the system is configured to send business logic data to the user interface through the intermediate layer (figures 1 and 2).

With respect to claim 11, Zhou teaches a computer-implemented method comprising:  
receiving multiple data entries in a user interface (figure 1, abstract, 0021; requests);  
passing the object to an intermediate layer (paragraphs 0031-0032), the intermediate layer being configured to interact with the user interface (figure 1; requests/replies and figure 2), the object model controller (drawing references 220, 222, and figure 2), and a layer of business logic (204);

performing one or more operations on data in the object passed to the intermediate layer (e.g. translating, paragraph 0033);

sending any one of data and instructions from the intermediate layer to the layer of business logic (paragraph 0033);

processing one or both of the data and instructions in the layer of business logic (drawing reference 204); and

sending one or both of processed data and processed instructions from the layer of business logic to the user interface (figure 1).

With respect to claim 12, Zhou teaches the computer-implemented method of claim 11 wherein the sending of one or both of processed data and process instructions comprises passing the one or both of the process data and processed instructions through the intermediate layer (paragraphs 0031-0032).

With respect to claim 15, Zhou teaches the computer-implemented method of claim 11 wherein the intermediate layer is configured to perform the following operations:

receiving an instruction from the object model controller (0032);  
performing one or more operations relating to the received instruction (0032-0033); and  
issuing one or more instructions to the layer of business logic (drawing reference 204).

With respect to claim 16 Zhou teaches the computer-implemented method of claim 15 wherein the intermediate layer determines whether the received instruction from the object model controller comprises any one of a known object, an unknown object, or a modification of a known object (paragraph 0090 and figure 5).

With respect to claim 17, Zhou teaches the computer-implemented method of claim 16 wherein, in response to the received instruction from the object model controller, the intermediate layer is further configured to perform any of the following operations: instructing

the layer of business logic to approve previous instructions and data entries; instructing the layer of business logic to save data in a database (figure 5); and initializing a framework to enable a user to perform data entry (figures 4-5 and paragraph 0091).

With respect to claim 18, Zhou teaches the computer-implemented method of claim 11 further comprising:

sending the data from the layer of business logic to a database (figure 5); and

saving the data in the database upon receiving the data from the layer of business logic (figure 5 drawing reference 510).

With respect to claim 19, Zhou teaches the computer-implemented method of claim 11 wherein the intermediate layer is configured to optimize one or more processes in the layer of business logic (paragraphs 0031-0032), and wherein the intermediate layer enables batch processing of data entered in the user interface (0074 and 0085; resource bundle).

With respect to claim 20, Zhou teaches the computer-implemented method of claim 11 wherein the intermediate layer maintains data entries and modifications among various object classes, and wherein the layer of business logic comprises common business functions and applications (paragraph 0036).

With respect to claim 21, Zhou teaches The computer-implemented method of claim 11 wherein a data flow between the user interface and the layer of business logic is initiated by one



or more actions of the user interface, wherein the one or more actions of the user interface comprise any one of an opening of the user interface and a data entry in the user interface (figure 4, drawing references 400, 406, 408).

With respect to claim 22, Zhou teaches an article comprising a machine-readable storage medium storing instructions operable to cause a machine to perform operations comprising:

receiving multiple data entries in a user interface (figure 1, abstract, 0021; requests);

in an object model controller (drawing references 220, 222, and figure 2), associating, at runtime (abstract), the multiple data entries with an object belonging to a model class (paragraphs 0036-0038), and preventing the object from being modified by another user interface (0102);

passing the data object to an intermediate layer (paragraphs 0031-0032), the intermediate layer being configured to interact with the user interface (figure 1; requests/replies and figure 2), the object model controller and a layer of business logic (204);

performing one or more operations on data in the object passed to the intermediate layer (e.g. translating, paragraph 0033);

sending one or both of data and instructions from the intermediate layer to the layer of business logic (paragraph 0033);

processing one or both of the data and instructions in the layer of business logic (drawing reference 204); and

sending one or both of processed data and processed instructions from the layer of business logic to the user interface (figure 1), wherein the sending of any one of processed data

and processed instructions comprises passing the any one of processed data and processed instructions through the intermediate layer (paragraph 0064 and figure 1).

With respect to claim 23, Zhou teaches a computing system comprising:

a network of computers (figure 1), wherein the network of computers comprises a database (figure 1, drawing reference 108) and at least one user interface (drawing reference 102) that is configured to collect multiple data entries from a corresponding computer in the network of computers (figure 1; requests/replies and abstract and paragraph 0029);

an object model controller configured to associate (drawing references 220, 222, and figure 2), at runtime (abstract), the multiple data entries with an object belonging to a model class (paragraphs 0036-0038), and to prevent the object from being modified by another user interface (0102);

a plurality of business logic configured to perform a plurality of business functions and applications (drawing reference 204); and

an intermediate layer (drawing reference 202 and figure 2) interacting with the at least one user interface (paragraph 0033), the object model controller and the plurality of business logic (system 110),

wherein the object model controller (drawing references 220, 222, and figure 2) is configured to provide the object with which the multiple data entries are associated to the intermediate layer, wherein the intermediate layer is configured to format and rearrange data in the object to optimize the processing of data in the plurality of business logic (paragraphs 0031-0033), wherein a data flow between the at least one user interface and the plurality of business

logic is conducted through the intermediate layer (figure 2 and 3 and paragraphs 0038 and 0057), wherein the plurality of business logic (drawing reference 204) is further configured to process data in the object and wherein the plurality of business logic interacts with the database (paragraph 0026 and drawing references 108 424, and 432).

With respect to claim 24, Zhou teaches A computing system comprising:

a user interface (102 (1-n)) configured to collect multiple data entries from a corresponding user device (figure 1, abstract, 0021; requests)

an object model controller (drawing references 220, 222, and figure 2) configured to associate, at runtime (abstract), the multiple data entries (figure 1; requests) with an object belonging to a model class (paragraphs 0036-0038), and to prevent the object from being modified by another user interface (0102);

business logic (drawing reference 204, and paragraphs 0030-0034) configured to process data objects belonging to the two or more corresponding model class classes (paragraphs 0036-0038, and resource bundle, 422); and

an intermediate layer (drawing reference 202 and figure 2) interposed between the user interface (at least drawing references 110, 212 and figures 1-2) and object model controller (drawing references 220, 222, and figure 2), and the business logic (drawing reference 204);

wherein the object model controller (drawing references 220, 222, and figure 2) is configured to provide the two or more data objects (resource bundle 422 with associated entries in web page 404) with which the multiple data entries are associated to the intermediate layer (drawing reference 202 and figure 2), and wherein the intermediate layer (drawing reference 202

Art Unit: 2167

and figure 2) is configured to buffer data (figure 4, drawing reference 424 and 0063) that is collected from the user interface (406) and associated with the two or more data objects (resource bundle 422), rearrange data in the two or more corresponding data objects (i.e. translate) into a format that is optimized to be processed by the business logic (0031), and provide the two or more data objects (resource bundle 422) whose data has been rearranged to the business logic for batch processing (0063).

With respect to claim 25, Zhou teaches the computing system of claim 24, wherein the intermediate layer (drawing reference 202 and figure 2) is further configured to request additional data (0055 and 0102 discusses adding additional content) from the user interface (102 (1-n)) to associate with at least one of the two or more data objects (resource bundle 422) following collection of the multiple data entries and association of the multiple data entries with the two or more data objects (resource bundle 422) but prior to provision of the two or more data objects to the business logic for processing (0102 discusses added additional content is translated (i.e. after being collected but before being distributed to the business logic).

### ***Response to Arguments***

Applicant's arguments filed 8/1/2007 have been fully considered but they are not persuasive.

With respect to claim 1, Applicant argues (p. 11 of response) that Zhou does not disclose “configuring, at runtime, multiple data entries with an object belonging to a model class” and

provide the object with which the multiple data entries are associated to an intermediate layer.”

The Examiner respectfully disagrees given the following:

In a system of internationalizing an application (abstract), Zhou teaches receiving client requests (0029, i.e. entries) to be translated into an appropriate form processed by the business logic (0033). The Zhou reference processes the received requests and constructs them in a variety of ways. In one example, the requests are constructed conforming to an execution model employing an interaction-based definition, which is based on a request-response model. Each interaction definition includes one or more command definitions whose functionality may be represented by an object. Therefore, the requests of the client are received and processed into a command definition represented by an object. Further, Zhou’s system uses an interaction based model employing command beans based on Java Beans (i.e. Java is an object oriented programming language) to associate requests to objects (paragraph, 0037).

Second, Zhou discloses that the requests that are entered are received by framework 220 (mapped to Applicant’s object model controller) (Zhou, paragraph 0032). The execution environment 202 (i.e. mapped to Applicant’s intermediate layer) supports receiving requests (i.e. from frame work 220) in one format. Therefore, it is respectfully submitted that the client requests are directly provided to the framework 220 and not the execution environment 202.

Another example of how Zhou teaches the limitations of claim 1 is presented in figure 4, which depicts Zhou’s compilation and translation system (Zhou, 0064). In figure 4 (coinciding with flow chart in figure 5), Zhou shows that a simple request in the form of a webpage (i.e. an initial logon page, 0056) is submitted to the system (drawing reference 404 of figure 4). Upon submission, elements (i.e. entries) from the page are extracted (i.e. collected, 0074) and

Art Unit: 2167

associated in a resource bundle (i.e. an object) pertaining to a locale (i.e. model class; for example, an English locale, 0066).

Applicant argues (p. 11 of response) that Zhou does not disclose, “an object model controller configured to... prevent the object from being modified by another user interface.” The Examiner respectfully disagrees. In the explanation above of Zhou collects entries from a submitted request, and associates the requests into a resource bundle (object). Once the resource bundle is created, it is stored in a repository (drawing reference 424). Zhou also discloses an aspect in their multi-layer architecture (figure 2) that contains an authentication model (270) and a security policy enforcement module. Zhou describes these components to authenticate users to the system (0048). Specifically, Zhou states prior to processing any requests clients and/or users are authenticated (lines 1-3 of 0048). As reasonably interpreted, Zhou enforces security rules to prevent an object from being modified by another user. Put another way, a specific user would have to be allowed to access the system to modify the content in a resource bundle (modifying content is disclosed in 0102)) thus preventing unauthorized users from modifying the content.

With respect to claim 4, Applicant argues (p. 13 of response) that Zhou does not disclose an intermediate layer that is “configured to optimize the arrangement of data for the business logic, wherein the rearrangement of data collected by the user interface comprises data collection from the user interface. The Examiner respectfully disagrees because again, in reference to figure 4, content from a locale specific web page (i.e. request) is submitted, and elements are extracted (0074), rearranged (drawing references 408, 426 and 422), and collected (i.e. data

Art Unit: 2167

collection, 0076) to form a resource bundle. Further, figure 4 goes on to show that the resource bundle is translated (drawing reference 430 and 0086).

With respect to claim 5, Applicant argues (p. 14 of response) that Zhou does not disclose buffering of data flow between the user interface and the business logic. The Examiner respectfully disagrees because Zhou discloses their system forming a resource bundle from the extracted data and storing (i.e. buffering) them in repository (drawing reference 424, see figures 4 and 6) to then be translated (drawing reference 430). Further, Zhou discloses that generally, the process of compilation and translation occur offline and then during runtime serving requests (0063). From collecting the extracted content and storing the collected content (offline) to be later processed (i.e. during runtime), Zhou sufficiently teaches the buffering of data flow enables the system to perform batch processing of a plurality of business processes (e.g. translating and processing requests).

With respect to claim 7, Applicant argues (p. 15 of the response) that Zhou does not describe “collecting and formatting one or more classes of objects”. The Examiner respectfully disagrees because Zhou teaches receiving requests in one format and in response returning replies. In Zhou’s system, the requests are collected (i.e. a resource bundle that is an object) and translated and formatted to be in appropriate form for the business logic (Zhou, 0033). Because the collected information is now in appropriate form, it is inherent that the business processes in Zhou would be reduced. For example, Zhou discloses separating the business logic layer from having to understand how to obtain data from external resources (Zhou, 0038).

With respect to claim 9 (page 16 of response), Applicant argues that Zhou does not disclose an object model controller that is “configured to send data requests to the intermediate layer.” Specifically, applicant has asked for clarity on what is considered the object model controller and what is the intermediate layer. As presented in the Office Action of 5/2/2007, and in response to the argument of claim 1 above, the Examiner submits that these claimed limitations have been mapped accordingly. Further, the object model controller (220, Zhou) resides on intermediate layer (202, figure 2 of Zhou) that interfaces to the business logic layer (204, and paragraph 0031) and thus is interposed between the user interface and object model controller and the business logic as stated in claim 1.

### *Conclusion*

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.



**Contact Information**

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Robert M. Timblin whose telephone number is 571-272-5627. The examiner can normally be reached on M-F 8:00-4:30.


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John R. Cottingham can be reached on 571-272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Robert M. Timblin



Patent Examiner AU 2167  
10/11/2007



JOHN COTTINGHAM  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100